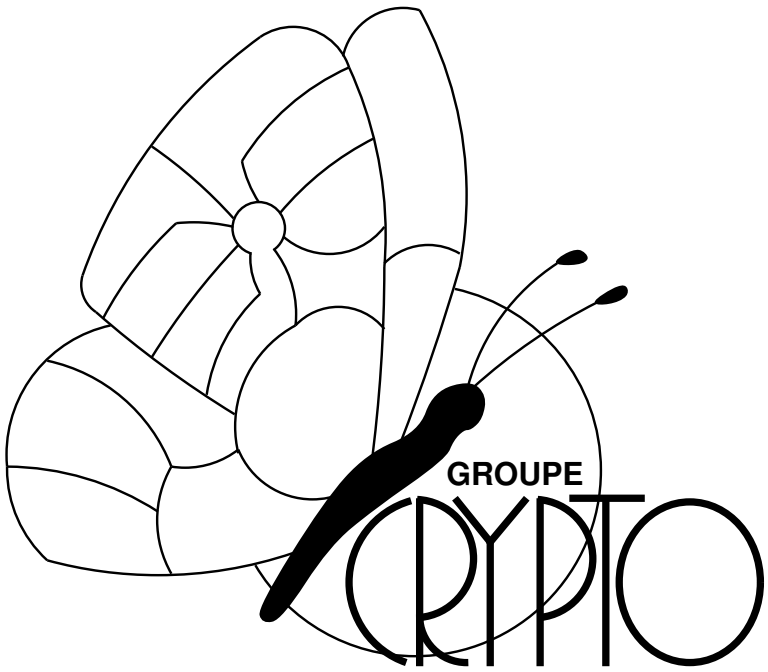
UCL
Université
catholique
de Louvain

REGARDS

UCL Crypto Group Technical Report Series

Short Private Exponent Attacks on Fast Variants of RSA

Mathieu Ciet · François Koeune · Fabien Laguillaumie · Jean-Jacques Quisquater



http://www.dice.ucl.ac.be/crypto/

Technical Report
CG–2003/4

Place du Levant 3
B-1348 Louvain-la-Neuve, Belgium

Phone: (+32) 10 472541
Fax: (+32) 10 472598

Short Private Exponent Attacks on Fast Variants of RSA

Mathieu Ciet[1)] François Koeune[1)]
Fabien Laguillaumie[2)] Jean-Jacques Quisquater[1)]

September 2002

[1)] UCL Crypto Group
Place du Levant, 3. 1348 Louvain-la-Neuve, Belgium
{ciet, fkoeune, jjq}@dice.ucl.ac.be – http://www.dice.ucl.ac.be/crypto/

[2)] France Télécom R&D
42 rue des Coutures, B.P. 6243, 14066 Caen Cedex 4, France,
fabien.laguillaumie@francetelecom.com – http://www.rd.francetelecom.fr/

Abstract. In this report, we study the adaptation of existing attacks on short private exponent on fast variants of the well-known RSA public-key cryptosystem, namely the RSA Multiprime and the Takagi family cryptosystems. The first one consists in a variant whose modulus is made up with strictly more than two primes, which permits to quickly decipher or sign using the Chinese Remainder Theorem. The second scheme has been introduced by Takagi in [21] and generalized by Lim, Kim, Yie and Lee, in [23]. A fast algorithm, involving some n-adic expansion of the modulus of the form $p^r q^s$, permits the decryption process to be very efficient. The use of short secret exponent may increase decryption or signature, but must be balanced with the risk to give rise to some powerful attacks, namely Wiener's continued fraction algorithm and Boneh-Durfee's methods. We study these attacks applied on the two fast variants of RSA.

Keywords. RSA type cryptosystem, low exponent attack, RSA Multiprime, $p^r q^s$ modulus, Takagi family scheme.

1 Introduction

One of the main drawbacks in the use of the RSA public-key cryptosystem [16] is the fact that encryption, decryption, signature generation or verification are quite slow, due to the size of the numbers and the exponentiation

CG–2003/4

operations involved. For some devices, like for example a smart card that generates RSA signatures, it is interesting to be able to do quickly the operations involving the private key. To accelerate this process one might use a short secret exponent. Unfortunately, Wiener showed in 1990 [25] that the secret exponent d can be found in polynomial time by using a continued fractions algorithm if (for a two-factors modulus N), $d < N^{1/4}$. Since that time, the most significant improvement concerning the attacks on RSA with low secret exponent is due to Boneh and Durfee in 1999 [2]. They recover d in polynomial time, if $d < N^{0.292}$.

The RSA Multiprime is composed of a modulus N made up with at least three prime factors: $N = p_1 p_2 \dots p_r$ with $r \geq 3$. The encryption process is the same as the classical RSA, but decryption and signature generation are performed by using Chinese Remainder Theorem (CRT) which speeds up these operations. Moreover parallel computation can be performed with r exponentiators.

At Crypto 1998, Takagi proposed a new public key cryptosystem [21], that is a modification of the well known RSA public key cryptosystem. He presents an interesting method that permits to speedup the decryption part when a specific modulus of the form $p^r q$ is used. A fast algorithm is proposed to retrieve $m \pmod{p^r}$: its running time is essentially that for computing $c^d \pmod{p}$, where m is the plaintext and c the ciphertext, by considering the p-adic expansion of $m \pmod{p^r}$: $m \pmod{p^r} = m_0 + m_1 p + m_2 p^2 + \ldots + m_{r-1} p^{r-1}$ where $m_0, m_1, m_2, \ldots, m_{r-1} \in [0, p-1]$. The CRT is then used to recover $m \pmod{p^r q}$ from $m \pmod{p^r}$ and $m \pmod{q}$. In [23], Takagi's public key cryptosystem was extended to moduli of the form $p^r q^s$.

Another method to accelerate the secret key operations – that could be used in conjunction with the previous one – is to use short secret exponents. In this report we analyze the impact of the attacks on short secret exponent against RSA MultiPrime and Takagi family scheme.

This report is organized as follows: after presenting some facts on RSA MultiPrime and generalized Takagi schemes in section 2, Wiener's attack is extended to those schemes in section 3. Next, in section 4 we consider Boneh-Durfee's attack in its basic and improved (using geometrically progressive matrices) form and give an upper bound on the exponent for which the attacks can be applied, and discuss corrections suggested by Hinek et al. in [10].

2 Mathematical Background

In this section, we briefly recall some facts on RSA MultiPrime and Takagi family schemes. This section may be safely skipped by the reader who already knows those schemes. The rest of this report is self-contained and can be understood without this part.

2.1 RSA MultiPrime

Compaq patented the RSA MultiPrime in January 1997. This technology consists in using an RSA modulus N with at least three non-equal prime factors:

$$N = p_1 p_2 \dots p_r \text{ with } r \geq 3 \ . \tag{1}$$

The encryption process is the same as the one in the classical RSA, but decryption and signature generation are performed by using the Chinese Remainder Theorem (CRT), which speeds up these operations. Moreover, parallel computation can be performed with r exponentiators. Figure 1 describes the recursive CRT algorithm for recovering the plaintext message m from the ciphertext $c = m^e \pmod N$, with $N = p_1 p_2 \dots p_r$.

- Input: c, d, p_i $(i = 1, \dots, r)$
- Output: m
  - $q_1 = 1$
  - for $i = 2, \dots, r$: $q_i = q_{i-1} p_{i-1}$, $u_i = q_i^{-1} \pmod{p_i}$
  - for $i = 1, \dots, r$: $d_i = d \pmod{(p_i - 1)}$, $c_i = c \pmod{p_i}$, $m_i = c_i^{d_i} \pmod{p_i}$.
  - $y_1 = m_1$
  - for $i = 2, \dots, r$: $y_i = y_{i-1} + q_i \times ((m_i - y_{i-1}) \times u_i \pmod{p_i})$
  - $m = y_r$

Figure 1: CRT algorithm for deciphering

We recall that the number of binary operations to compute $x^d \pmod N$ is approximately $\frac{3}{2}|d| \times |N|^2$ where $|.|$ denotes the number of bits. Table 1 shows the number of binary operations[1] required to decipher, using the classical RSA and the RSA MultiPrime.

[1]The final step of reconstitution is negligible and has therefore been left out.

Table 1: Number of binary operations in RSA decipherment

	Parallel computation	Number of operations	
RSA	no	$\frac{3}{2}\|N\|\|N\|^2$	$= \frac{3}{2}\|N\|^3$
RSA with CRT	no	$\frac{2.3}{2}\left(\frac{\|N\|}{2}\right)\left(\frac{\|N\|}{2}\right)^2$	$= \frac{3}{8}\|N\|^3$
RSA with CRT	yes	$\frac{3}{2}\left(\frac{\|N\|}{2}\right)\left(\frac{\|N\|}{2}\right)^2$	$= \frac{3}{16}\|N\|^3$
MultiPrime r-factor	no	$\frac{r.3}{2}\left(\frac{\|N\|}{r}\right)\left(\frac{\|N\|}{r}\right)^2$	$= \frac{3}{2r^2}\|N\|^3$
MultiPrime r-factor	yes	$\frac{3}{2}\left(\frac{\|N\|}{r}\right)\left(\frac{\|N\|}{r}\right)^2$	$= \frac{3}{2r^3}\|N\|^3$

On a security point of view, the maximum number of prime factors allowed in the modulus is determined by the point of intersection of the curves representing the time necessary for the Number Field Sieve (NFS) and Elliptic Curve Method (ECM) algorithms to factorize a fixed size modulus depending on the number of prime factors. The complexity of the first algorithm grows with the size of the number to be factorized. The second one increases with the size of the prime factors in the number. These two methods have a sub-exponential complexity: $L_N[\frac{1}{3}, c]$ with c a small constant for NFS, and $L_p[\frac{1}{2}, \sqrt{2}]$ for ECM if p is a factor of N. This criterion leads to Table 2. Recall that

$$L_N[\alpha, c] = \mathcal{O}(e^{(c+o(1))(\ln N)^{\alpha}(\ln\ln N)^{1-\alpha}}) \tag{2}$$

with c a small positive constant and α a real such as $0 < \alpha < 1$.

For instance, using a modulus with 3 prime factors makes it possible to generate an RSA MultiPrime signature 9/4 times faster than with the classical RSA without parallel computation, and 27/8 times faster with RSA with parallel computation. Below we will see that decreasing the size of d accelerates this generation even more.

2.2 Takagi family schemes

In this section, we briefly introduce the Takagi family schemes and give some facts on the decryption part that is the main point of these schemes.

The $p^k q$ cryptosystem was first introduced by Takagi in [21], and generalized by Lim, Kim, Yie and Lee, in [23] to moduli of the form $p^r q^s$. The moduli are appropriately chosen to resist factoring algorithms such as the

Table 2: Optimal number of prime factors for a specific modulus size (cf. [26])

Modulus size in bits	Number of primes
512	2
1024	3
1536	3
2048	3
2560	3
3072	3
3584	3
4096	4
8192	5

Number Field Sieve and Elliptic Curve Method (NFS and ECM respectively for short).

The private and public parameters d and e are chosen as to verify the following equation

$$ed \equiv 1 \pmod{\operatorname{lcm}(p-1, q-1)}$$

thus the secret exponent d is much smaller than the modulus n. The encryption is done as for the classical RSA, namely, for a message m, by computing $m^e \pmod{n}$, and the decryption part is decomposed in two phases: firstly recovering $m \pmod{p^r}$ and $m \pmod{q^s}$ (details are given below) and secondly $m \pmod{n}$ using the Chinese Remainder Theorem proposed by Quisquater and Couvreur in [15]. To recover the exact value of $m \pmod{p^r}$, the algorithm based on p-adic expansion introduced in [20] is used, the same holds to recover the value of $m \pmod{q^s}$.

Let c_p be the ciphertext m^e reduced modulo p^r, and m_p be the plaintext modulo p^r, related to c_p by the following relationship

$$c_p = m_p^e \pmod{p^r} \ .$$

Given the ciphertext c, the value of m_p is recovered using the p-adic expansion of m_p

$$m_p \equiv \sum_{h=0}^{r-1} p^h \, K_h \pmod{p^r} \ . \tag{3}$$

For $i \in \{0, \ldots, r-1\}$, let

$$F_i(X_0, \ldots, X_i) = \left(\sum_{h=0}^{i} p^h X_h \right)^e \quad \text{and} \quad G_i(X_0, \ldots, X_i) = e \left(\sum_{h=0}^{i} p^h X_h \right)^{e-1} ,$$

we have

$$F_i \equiv F_{i-1} + p^i G_{i-1} X_i \pmod{p^{i+1}} . \tag{4}$$

The values K_h of (3) are recursively recovered as *the* solution in X_h of the following equation

$$c_p \equiv F_{h-1} + p^h G_{h-1} X_h \pmod{p^{h+1}} . \tag{5}$$

The same is applied to recover $m \pmod{q^s}$, and m is finally obtained using the Chinese Reminder Theorem applied on $m \pmod{p^r}$ and $m \pmod{q^s}$. For further details concerning the running time of the decryption process, we refer the reader to [22].

3 Wiener's Attack.

The first important result about the attack over RSA with short secret exponent is due to Wiener in 1990 [25]. Continued fractions algorithm is used to find a fraction involving the secret exponent from a fraction totally determined by the public key. The starting point of this attack, and more generally of all attacks on short secret exponent, is the following equation:

$$ed \equiv 1 \pmod{\varphi(N)} . \tag{6}$$

where $\varphi(.)$ is the Euler totient function, e the private exponent and d the public exponent. Wiener exploits the fact that the fraction $\frac{e}{pq}$, that is totally determined by public parameters, is an approximation of a fraction whose denominator is a multiple of d.

Theorem 1 (cf. Theorem 184 of [9]). *Suppose that* $\gcd(a, b) = \gcd(c, d) = 1$ *and*

$$\left| \frac{a}{b} - \frac{c}{d} \right| \leq \frac{1}{2d^2} .$$

Then c/d is one of the convergents of the continued fraction expansion of a/b.

3.1 Multiprime case

Let us consider the case of RSA Multiprime with a modulus N of r distinct prime factors $N = p_1 \dots p_r$ and suppose that $\gcd(p_i - 1, p_j - 1) = 2$, for all $i \neq j$, to combat factorization methods. Therefore, $\mathrm{lcm}(p_1 - 1, \dots, p_r - 1) = (p_1 - 1) \dots (p_r - 1)/2^{r-1}$. Without restriction we can suppose that $p_1 < p_2 < \dots < p_r < 2p_1$ and consequently,

$$p_1^r < N < 2^{r-1} p_1^r \ . \tag{7}$$

From the equation $ed \equiv 1 \pmod{\mathrm{lcm}(p_1 - 1, \dots, p_r - 1)}$ we deduce the existence of $k \in \mathbb{Z}$, odd, such that

$$ed = 1 + \frac{k}{g}(p_1 - 1) \dots (p_r - 1) \ . \tag{8}$$

with $g = 2^{r'},\ r' \leq r - 1$. Equation (8) can be rewritten as

$$ed = 1 + \frac{k}{g}\varphi(N) \ . \tag{9}$$

Then to apply Theorem 1, the following condition is needed

$$\left| \frac{e}{N} - \frac{k}{dg} \right| < \frac{1}{2(dg)^2} \ ,$$

and after simplification, we obtain that the attack is possible if

$$\frac{2^{(r-1)/r} r}{g N^{1/r}} < \frac{1}{2(dg)^2} \quad \text{i.e.} \quad d < \frac{N^{1/2r}}{2^{(2r-1)/2r} \sqrt{rg}} \ .$$

Therefore, the adaptation of Wiener's attack to RSA MultiPrime with r factors succeeds if d is approximately less than $N^{1/2r}$. There remains to identify the correct convergent among all. Here is a test to find out $\frac{k}{dg}$. First of all, as $ed > N$, the inequality $k > g$ holds. So, for each convergent computed, the Euclidean division of edg by k leads to guesses for $\varphi(N)$ and g. If the guesses are true, calculating dg/g gives the secret exponent d. We check whether it decrypts a ciphertext $c = m^e \pmod N$ previously computed with an arbitrary message $m \in \mathbb{Z}/N\mathbb{Z}$. If the message m is not recovered, we perform this test for the following convergent, until d is found. This allows to recover the factorization of the modulus, using the following result:

Lemma 2. *Let $N = p_1 \dots p_s$ an integer. The knowledge of a multiple of $\varphi(N)$ gives a probabilistic polynomial algorithm which factorizes N.*

Proof. Let $N = p_1 \dots p_s$, with $p_i - 1 = 2^{k_i} q_i$ for each $i = 1 \dots s$. Moreover, the p_is are ordered so that $k_1 \leq \dots \leq k_s$. We write $\alpha\varphi(N) = 2^k r$ a multiple of $\varphi(N)$, with r odd. Let $m \in \mathbb{Z}/N\mathbb{Z}$ such that $\gcd(m, N) = 1$. The three following facts may happen:

- $m^r \equiv 1 \pmod{N}$,
- $\exists i \in \mathbb{N}$, $0 \leq i < k$ such that $m^{2^i r} \equiv -1 \pmod{N}$,
- $\exists i \in \mathbb{N}$, $0 \leq i < k$ such that $m^{2^i r} \not\equiv -1 \pmod{N}$ and $m^{2^{i+1} r} \equiv 1 \pmod{N}$.

When the last case happens, $m^{2^i r}$ is a square root of 1 modulo N, distinct from ± 1, so $(m^{2^i r} - 1)$ and $(m^{2^i r} - 1)$ are non trivial divisors of zero. Computing their gcd with N gives a non-trivial divisor h of N. We can perform this operation again with N/h or h as $\varphi(N)$ is still a multiple of $\varphi(N/h)$ or $\varphi(h)$ because $\varphi(N) = \varphi(N/h)\varphi(h)$. There remains to evaluate the probability to find a m that allows to factorize. First, define

$$B(n) := \{m \in (\mathbb{Z}/n\mathbb{Z})^* : m^r \equiv 1 \pmod{n} \text{ or } \exists i \in \mathbb{N}, 0 \leq i < k : m^{2^i r} \equiv -1 \pmod{n}\} .$$

$$P(n) := \{m \in (\mathbb{Z}/n\mathbb{Z})^* : m^r \equiv 1 \pmod{n}\} ,$$

and

$$B_j(n) := \{m \in (\mathbb{Z}/n\mathbb{Z})^* : m^{2^j r} \equiv -1 \pmod{n}\} ,$$

for $0 \leq j \leq k - 1$, so as

$$B(n) = P(n) \cup \bigcup_{0 \leq j \leq k-1} B_j(n) .$$

Using the Chinese Remainder Theorem, we can evaluate $\sharp P(n) = \prod_{i=1}^{s} \sharp P(p_i)$, and since $\sharp P(p_i) = \gcd(r, p_i - 1)$, we have

$$\sharp P(n) = \prod_{i=1}^{s} \gcd(r, p_i - 1) .$$

Now let us consider $Q_j(n)$. As before, $\sharp Q_j(n) = \prod_{i=1}^{s} \sharp Q_j(p_i)$. Note that

$$\begin{aligned} Q_j(p_i) \neq \emptyset &\iff (-1)^{\frac{p_i - 1}{\gcd(2^j r, p_i - 1)}} = 1 \\ &\iff \frac{2^{k_i} q_i}{2^{\inf(j, k_i)} \gcd(r, q_i)} \text{ is even} \\ &\iff j < k \end{aligned}$$

and so $\sharp Q_j(n) = \gcd(2^j r, p_i - 1) = 2^j \gcd(r, q_i)$. Finally, as we have arranged the $p_i s$ we obtain

$$\sharp Q_j(p_i) = \begin{cases} 0 & \text{if } j \geq k_1 \\ 2^{js} \prod_{i=1}^{s} \gcd(r, q_i) & \text{if } j < k_1 \end{cases}$$

Then $\sharp B(n) = \sharp P(n) + \sum_{j=0}^{k-1} \sharp Q_j(n) = \left(1 + \sum_{j=0}^{k-1}\right) \prod_{i=1}^{s} \gcd(r, q_i)$.
While noticing that $k_1 \leq k$, we obtain after a routine calculus

$$\frac{\sharp B(n)}{\varphi(N)} \leq \frac{1}{2^{s-1}}.$$

This means that the probability of finding a m which allows the factorization is greater than $1 - 1/2^{s-1}$ □

3.2 Takagi family schemes

In the case of Takagi family schemes the public modulus has a particular form $N = p^r q^s$, with p and q two large prime numbers. Moreover the exponents are generated using the following equation:

$$ed \equiv 1 \pmod{\operatorname{lcm}(p-1, q-1)}. \tag{10}$$

This generation seems to prevent a direct application of Wiener's attack. However, another possibility would be for the attacker to find d' such that $ed' \equiv 1 \pmod{\varphi(N)}$ where $\varphi(N) = p^{r-1} q^{s-1}(p-1)(q-1)$, the attack will work and the attacker will be able to recover the plaintext. Indeed, there is an integer k such that $ed' = 1 + k\varphi(N)$ and then

$$\left| \frac{e}{N} - \frac{k}{d'} \right| = \left| \frac{1}{Nd'} + \frac{k}{d'} \left(\frac{\varphi(N)}{N} - 1 \right) \right| \ . \tag{11}$$

We have

$$\left| \frac{\varphi(N)}{N} - 1 \right| = \frac{1}{p} + \frac{1}{q} - \frac{1}{pq} \ .$$

Moreover, we suppose that $p < q < 2p$, which means that the prime factors in the modulus have the same size. Then

$$\left| \frac{\varphi(N)}{N} - 1 \right| \leq \frac{2}{p} \quad \text{and} \quad N < 2^s p^{r+s} \ ,$$

thus

$$p > \frac{N^{1/(r+s)}}{2^{s/(r+s)}} \ .$$

To mount the attack, we need

$$\left|\frac{\varphi(N)}{N} - 1\right| < \frac{1}{(d')^2} \ .$$

After simplification, we obtain that the attack is possible if

$$d' < N^{\frac{1}{2(r+s)}} \ . \tag{12}$$

However, since $e < \text{lcm}(p-1)(q-1)$, it is easy to see that the two conditions given by equations (10) and (12) are mutually exclusive. Wiener's attack is therefore not applicable.

4 Lattice Attacks

4.1 Adaptation of the lattice approach

In [2], Boneh and Durfee improved Wiener's bound by using a lattice reduction approach.

4.1.1 Multiprime case

We focus on an r-factor MultiPrime and examine the impact and the problems when adapting the Boneh-Durfee lattice attack. The equality $ed + k\varphi(N) = 1$ holds with $\varphi(N) = (p_1 - 1)\dots(p_r - 1)$. We define

$$A := N + (-1)^r \quad \text{and} \quad s := \varphi(N) - N - (-1)^r \ .$$

and have the following equation:

$$k(A + s) \equiv 1 \pmod{e} \ . \tag{13}$$

Without restriction we can suppose that: e is of the same order as N, $d < N^\delta$ and $p_1 < \ldots < p_r < 2p_1$. Then

$$|k| = \frac{ed-1}{\varphi(N)} \leq \frac{ed}{\varphi(N)} \leq \frac{ed}{N} < e^\delta, \tag{14}$$

and

$$|s| < \sum_{i=1}^{r} \frac{N}{p_i} < r\frac{N}{p_1} < r2^{\frac{1}{r}-1}N^{1-\frac{1}{r}} \ . \tag{15}$$

Let $f(x, y) = x(A+y)-1$ be a bivariate polynomial with integer coefficients. The problem is then to find $(x_0, y_0) \in \mathbb{Z}^2$ such that:

$$f(x_0, y_0) \equiv 0 \pmod{e} \quad \text{with} \quad |x_0| < e^{\delta} =: X \quad \text{and} \quad |y_0| < e^{1-\frac{1}{r}} =: Y \ . \tag{16}$$

Let us define a norm over polynomials $P(x, y) = \sum_{i,j} a_{i,j} x^i y^j \in \mathbb{Z}[x, y]$ as:

$$\|P(x, y)\|^2 = \sum_{i,j} a_{i,j}^2 \ . \tag{17}$$

Boneh-Durfee's approach is based on the following theorem.

Theorem 3 cf. [11]. *Let $P(x, y)$ be a polynomial which is a sum of at most w monomials. Suppose that $P(x_0, y_0) \equiv 0 \pmod{e^m}$ for some positive integer m, where $|x_0| < X$ and $|y_0| < Y$. If $\|P(xX, yY)\| < e^m/\sqrt{w}$, then $P(x_0, y_0) = 0$ holds over the integers.*

The starting point of this attack is the polynomial $f(x, y) := x(A+y)-1$, for which (k, s) is a solution modulo e. The main idea is to find two polynomials of low norm, using LLL, that have (k, s) as a solution modulo e^m, to apply Theorem 3. Then, we exploit the fact that the solutions live in $\mathbb{Z}$ to recover them while computing a resultant of the two polynomials.

As far as the low norm polynomial is concerned, Boneh and Durfee construct these two polynomial families:

$$g_{i,k}(x, y) := x^i f^k(x, y) e^{m-k} \quad \text{and} \quad h_{j,k}(x, y) := y^j f^k(x, y) e^{m-k} \ .$$

We note that (k, s) is a solution modulo e^m of all these polynomials, for $k = 0, \ldots, m$. The aim is to find a low norm integer linear combination of the polynomials $g_{i,k}(xX, yY)$ (called x-shifts) and $h_{j,k}(xX, yY)$ (called y-shifts). Thus, a lattice is built using the coefficients of these polynomials. For each $k = 0, \ldots, m$, Boneh and Durfee use $g_{i,k}(xX, yY)$ for $i = 0, \ldots, m-k$, and $h_{j,k}(xX, yY)$ for $j = 0, \ldots, t$ to construct a matrix like the one in Figure 2. The two integers m and t are parameters which are optimized during the construction.

Now, we can use the very powerful LLL's lattice reduction algorithm to obtain low norm elements of this lattice. The size of these elements is bounded by the following well-known results [12].

Lemma 4. *Let L be a lattice and $(b_1, \ldots, b_d)$ be an LLL-reduced basis of L. Then*

$$\|b_1\| \leq 2^{d/2} \det(L)^{1/d} \ . \tag{18}$$

	1	x	xy	x^2	x^2y	x^2y^2	x^3	x^3y	x^3y^2	x^3y^3	y	xy^2	x^2y^3	x^3y^4
e^3	e^3													
xe^3		e^3X												
fe^2	–	–	e^2XY											
x^2e^3				e^3X^2										
xfe^2		–		–	e^2X^2Y									
f^2e	–	–	–	–	–	eX^2Y^2								
x^3e^3							e^3X^3							
x^2fe^2				–			–	e^2X^3Y						
xf^2e		–		–	–		–	–	eX^3Y^2					
f^3	–	–	–	–	–	–	–	–	–	X^3Y^3				
ye^3											e^3Y			
yfe^2			–								–	e^2XY^2		
yf^2e			–		–	–					–	–	eX^2Y^3	
yf^3			–			–		–	–	–	–	–	–	X^3Y^4

Figure 2: The matrix spanned by $g_{i,k}$ and $h_{j,k}$ for $k = 0, \ldots, 3$, $i = 0, \ldots, 3-k$, and $j = 0, 1$. The '–' symbols denote non-zero entries whose value are not accounted for (cf. [2])

Lemma 5. *Let L be a lattice spanned by $(u_1, \ldots, u_d)$, and let $(b_1, \ldots, b_d)$ be the result of applying LLL to the given basis. Suppose that $\min_i \|u_i^*\| \geq 1$. Then*

$$\|b_2\| \leq 2^{d/2} \det(L)^{1/(d-1)} \quad . \tag{19}$$

With the two above lemmas and Boneh-Durfee's theorem we can find conditions on δ so that the norm of the first two basis vectors is small enough to have (x_0, y_0) as a solution over the integers. We can then recover s by computing the resultant $h(y) = \mathrm{Res}_x(g_1, g_2)$, and by finding its root over the integers. We are then able to compute $\varphi(N)$.

Remark. Boneh and Durfee [2] note that this attack is heuristic because nothing guarantees that $g_1(x, y)$ and $g_2(x, y)$ are algebraically independent. If it is not the case, the resultant is null and the factorization cannot be recovered. However, they also note that the attack works well in practice.

Being triangular, the determinant of the matrix is easy to compute:

$$\det(L) = \det_x \det_y \quad , \tag{20}$$

with

$$\begin{cases} \det_x = e^{\frac{m(m+1)(m+2)}{3}} X^{\frac{m(m+1)(m+2)}{3}} Y^{\frac{m(m+1)(m+2)}{6}} \\ \det_y = e^{\frac{tm(m+1)}{2}} X^{\frac{tm(m+1)}{2}} Y^{\frac{t(m+1)(m+t+1)}{2}} \quad . \end{cases} \tag{21}$$

The determinant $\det_x$ is the determinant of the sub-matrix only spanned by the x-shifts, and $\det_y$ is the product of the diagonal terms of the part of the matrix involving only the y-shifts.

Remark. If we consider the lattice corresponding to the matrix of the x-shifts, we theoretically recover Wiener's bound, but Blömer and May noticed in [1] that LLL always provides two algebraically dependant vectors in that case, which means that the attack is not effective.

The dimension of the matrix becomes $w = \frac{(m+1)(m+2)}{2} + t(m+1)$, and, substituting X and Y by their value in the whole lattice, we obtain:

$$\det(L) = e^{(\frac{\delta}{3}+\frac{1}{2}-\frac{1}{6r})m^3+(1+\frac{\delta}{2}-\frac{1}{2r})tm^2+(\frac{1}{2}-\frac{1}{2r})mt^2+O(m^3)} \ . \tag{22}$$

To apply Theorem 3 on the two first vectors produced by LLL. We have to find out the largest value for δ such that:

$$\det(L) < \frac{e^{m(w-1)}}{\gamma} \quad \text{with} \quad \gamma = (w2^w)^{\frac{w-1}{2}} \ . \tag{23}$$

By neglecting the low terms, and writing $w = m^2/2+tm+\mathcal{O}(m^2)$, we obtain:

$$\left(\frac{\delta}{3} - \frac{1}{6r}\right) m^3 + \left(\frac{\delta}{2} - \frac{1}{2r}\right) tm^2 + \left(\frac{1}{2} - \frac{1}{2r}\right) t^2 m < 0 \ . \tag{24}$$

For each m, the minimum is reached for $t = \dfrac{1-r\delta}{2(r-1)}m$. So we obtain:

$$m^3 \left(\frac{\delta}{3} - \frac{1}{6r} - \frac{(\delta r-1)}{2(r-1)}\left(\frac{\delta}{2} - \frac{1}{2r}\right) + \frac{(\delta r-1)^2}{4(r-1)^2}\left(\frac{1}{2} - \frac{1}{2r}\right)\right) < 0 \ . \tag{25}$$

and we study

$$\frac{-r}{8(r-1)}\delta^2 + \left(\frac{1}{3} + \frac{1}{4(r-1)}\right)\delta - \frac{1-4r}{24r(r-1)} < 0 \ . \tag{26}$$

and finally

$$\delta < \frac{4}{3} - \frac{1}{3r} - \frac{2}{3r}\sqrt{4r^2-5r+1} \ . \tag{27}$$

Hence we can obtain a polynomial $g_1(x, y) \in \mathbb{Z}[X, Y]$ which has (x_0, y_0) as a solution over the integers. We need another polynomial $g_2(x, y)$ which will be found as Lemma 5 holds. We obtain another polynomial with (x_0, y_0) as a solution. We recover y_0 when solving the resultant $h(y) = \mathrm{Res}_x(g_1, g_2) \in \mathbb{Z}[y]$, which allows us to compute $\varphi(N)$, and this scheme is broken. The bound

for our adaptation of Wiener's attack on 3-factor MultiPrime is improved: our general bound is equal to 0.1799 in this case. However, if the two polynomials g_1 and g_2 have a common factor, then $h(y)$ is identically null. In this sense, this attack is heuristic, but according to Boneh and Durfee, it works well in practice. By the way, using a secret exponent less than $N^{\frac{4}{3}-\frac{1}{3r}-\frac{2}{3r}\sqrt{4r^2-5r+1}}$ can be dangerous.

4.1.2 Takagi family scheme

In this section, we try to apply the attack to the Takagi family cryptosystems. Once again, we take into account the possibility for the attacker to find a d' such that $ed' \equiv 1 \pmod{\varphi(N)}$. For convenience, let us denote this d' by d. Starting from equation (6), we have, for some $k \in \mathbb{Z}$,

$$ed + k(p^{r-1}q^{s-1}(p-1)(q-1)) = 1.$$

Defining $A := N$ and $u := p^{r-1}q^{s-1} - p^r q^{s-1} - p^{r-1}q^s$, we can rewrite this as

$$k(A+u) \equiv 1 \pmod{e} \ .$$

Thus we need some upper bounds on k and u. If we suppose that e has the same size as N and that $d = N^\delta$, then

$$|k| = |\tfrac{1-ed}{\varphi(N)}| < e^\delta =: X \tag{28}$$

$$|u| < 2^s p^{r+s-1} < N^{\frac{r+s-1}{r+s}} =: Y \ . \tag{29}$$

We construct the same matrix than in the previous section. The determinant of the matrix becomes:

$$\det(L) = e^{\left(\frac{\delta}{3}+\frac{1}{3}-\frac{r+s-1}{6(r+s)}\right)m^3+\left(\frac{1}{2}+\frac{\delta}{2}+\frac{r+s-1}{2(r+s)}\right)tm^2+\left(\frac{r+s-1}{2(r+s)}\right)mt^2+o(m^3)} \ . \tag{30}$$

In order to apply Theorem 3 to the first two vectors of the reduced basis, we need to verify the condition of Lemma 5, which means that we have to search for the largest δ such that

$$\det(L) < \frac{1}{\gamma} e^{m(w-1)} \text{ with } \gamma = (w2^w)^{\frac{w-1}{2}}, \tag{31}$$

where

$$w = \frac{(m+1)(m+2)}{2} + t(m+1) = m^2/2 + tm + o(m^2) \ .$$

Neglecting some constant, equation (31) is equivalent to

$$\left(\frac{\delta}{3} - \frac{1}{6(r+s)}\right) m^3 + \left(\frac{\delta}{2} - \frac{1}{2(r+s)}\right) tm^2 + \left(\frac{1}{2} - \frac{1}{2(r+s)}\right) t^2 m < 0 \ .$$

For each m the value of t that minimizes this expression is

$$t = \frac{1 - \delta(r+s)}{2(r+s-1)} \ .$$

Substituting this value to t in the previous equation, a new equation in δ of degree two is obtained

$$-\frac{r+s}{8(r+s-1)}\delta^2 + \frac{2(r+s)-1}{6(r+s-1)}\delta + \frac{1-4(r+s)}{24(r+s-1)(r+s)} < 0,$$

which means that

$$\delta < \frac{4(r+s) - 1 - 2\sqrt{4(r+s)^2 - 5(r+s) + 1}}{3(r+s)}.$$

Therefore, the first two vectors, $g_1(x, y)$ and $g_2(x, y)$, given by LLL will have $(k,\ u)$ as a solution over the integers, and the resultant $h(y) := \mathrm{Res}_x(g_1, g_2)$ allows to recover u: once u is known, p and q can be recovered with high probability by computing $p + q = (u/\gcd(N, u)) + 1$ and $pq = N/\gcd(N, u)$. The attack therefore requires that there exists d', with $ed' \equiv 1 \pmod{\varphi(N)}$, such that

$$d' < N^{\frac{4(r+s)-1-2\sqrt{4(r+s)^2-5(r+s)+1}}{3(r+s)}}$$

with $N = p^r q^s$. Once again, such d cannot satisfy 10. Special case and numerical values are given in Appendix A.

Remark. Recently Hinek et al. [10] pointed out that, due to some careless approximation (we refer to their paper for details), Boneh-Durfee's bound was inaccurate and too optimistic. In Appendix A.1 we take their remarks into account to compute more precise and rigorous values for a bound on δ. By the way, since the LLL algorithm used to mount the attack is heuristic, it is important to notice that the constant can clearly be neglected. Indeed, in practice, the constants appear to be small enough to be ignored.

4.2 Improved bound using geometrically progressive matrices

In [2], Boneh and Durfee introduce the notion of *geometrically progressive matrices*. Rows and columns of the matrix M are divided into $a+1$ blocks of size b, for a, $b \in \mathbb{N}$. Rows (resp. columns) are indexed by pairs (i,j), with $i = 0, \ldots, a$ and $j = 0, \ldots, b$, so that the pair (i,j) corresponds to the $(bi+j)$th row (resp. column) of M. The element of the (i,j)th column and (k,l)th row is denoted $M(i,j,k,l)$. Geometrically progressive matrices are defined as follows.

Definition 6. Let $C, D, c_0, c_1, c_2, c_3, c_4, \beta$ be real numbers with $C, D, \beta \geq 1$. A matrix M is said to be *geometrically progressive with parameters* $(C, D, c_0, c_1, c_2,$
$c_3, c_4, \beta)$ if the following conditions hold for all $i, k = 0, \ldots, a$ and $j, l = 1, \ldots, b$:

1. $|M(i,j,k,l)| \leq C.D^{c_0+c_1 i+c_2 j+c_3 k+c_4 l}$,
2. $M(k,l,k,l) = D^{c_0+c_1 k+c_2 l+c_3 k+c_4 l}$,
3. $M(i,j,k,l) = 0$ whenever $i > k$ or $j > l$,
4. $\beta c_1 + c_3 \geq 0$ and $\beta c_2 + c_4 \geq 0$.

Boneh and Durfee proved the following theorem that gives a bound on the determinant of a geometrically progressive matrix, from which some rows have been removed.

Theorem 7. *Let M be an $(a+1)b \times (a+1)b$ geometrically progressive matrix with parameters $(C, D, c_0, c_1, c_2, c_3, c_4, \beta)$, and B a real number. Define*

$$S_B := \{(k,l) \in \{0, \ldots, a\} \times \{1, \ldots, b\} | M(k,l,k,l) \leq B\},$$

and set $w := |S_B|$. If L is the lattice defined by the rows $(k,l) \in S_B$ of M, then

$$\det(L) \leq ((a+1)b)^{w/2}(1+C)^{w^2} \prod_{(k,l)\in S_B} M(k,l,k,l).$$

The idea of this method is to remove some vectors of the basis, in particular those whose contribution in the determinant of the lattice is too big. The previous theorem permits to control the value of this determinant. As the following lemma states, the matrix, denoted M_y, and defined as the matrix made up with the rows corresponding to the y-shifts of the matrix M of Sect. 4.1, and whose columns correspond to the columns of this matrix such that $x^i y^j$ with $j > i$, is geometrically progressive.

4.2.1 Multiprime case

Lemma 8. *For all positive integers m, t, the matrix M_y is geometrically progressive with parameters $(m^{2m}, e, m, \delta + 1 - 1/r, -1/r, -1, 1, 2)$.*

Proof. To simplify, we take $e = N$. We recall that

$$h_{l,k}(xX, yY) = e^{m-k} y^l Y^l f^k(xX, yY) = \sum_{u=0}^{k} \sum_{v=0}^{u} c_{u,v} x^u y^{v+l},$$

with

$$c_{u,v} = \binom{k}{u} \binom{u}{v} (-1)^{k-u} e^{m-k} A^{u-v} X^u Y^{v+l}.$$

So we can compute

$$M_y(i, j, k, l) = c_{i,i+j-l} = \binom{k}{i} \binom{i}{i+j-l} (-1)^{k-i} e^{m-k} A^{l-j} X^i Y^{i+j}.$$

Condition (iii) of the definition is satisfied, and by replacing $X = e^{\delta}$, $Y = e^{1-\frac{1}{r}}$, and because $A = e$, we have:

$$|M_y(i, j, k, l)| \leq \binom{k}{i} \binom{i}{i+j-l} e^{m-k+l-j+\delta i+(1-1/r)(i+j)},$$

so

$$|M_y(i, j, k, l)| \leq m^{2m} e^{m+(\delta+1-\frac{1}{r})i-\frac{1}{r}j-k+l}.$$

We calculate

$$M_y(k, l, k, l) = e^{m+(\delta+1-\frac{1}{r})k-\frac{1}{r}l-k+l},$$

which satisfies condition (ii). Finally, as these two inequalities:

$$2(\delta + 1 - 1/r) - 1 \geq 0 \quad \text{and} \quad 2(-1/r) + 1 \geq 0$$

hold, our matrix M_y is a geometrically progressive matrix with parameters $(m^{2m}, e, m, \delta + 1 - \frac{1}{r}, -\frac{1}{r}, -1, 1, 2)$. □

The geometrically progressive matrix, noted M_1, is constructed on the basis of the matrix defined in Sect. 4.1, but taking twice as many y-shifts as in the previous one, i.e. setting $t = \frac{1-\delta r}{r-1}$. Then the rows corresponding to the y-shifts whose entry on the diagonal exceeds e^m are removed. Using Gaussian elimination, we obtain a unitary matrix A, such that $M_1 = AM_2$, with M_2 of the form:

	$1\ x\ xy\ \ldots\ x^m y^m$	$y\ y^2\ \ldots\ y^t$	$\ldots$	$x^m y^{m+1}\ \ldots\ x^m y^{m+t}$
x-shifts	Δ	0		
selected y-shifts	0	M'_y		

with Δ a diagonal matrix. So we can apply Theorem 7 on the lattice L_2, because $\det(L_1) = \det(L_2)$. Moreover, $\det(L_2) = \det(\Delta)\det(L'_y)$, since x-shifts and selected y-shifts are orthogonal. The dimension w of lattice L_2 is computed as follows:

$$w = m(m+1)(m+2)/2 + w',$$

where w' is the dimension of L'_y. The elements $M(k,l,k,l)$ that will be removed are those for which $M(k,l,k,l) < e^m$, i.e.

$$e^{m+(\delta-\frac{1}{r})k+(1-\frac{1}{r})l} < e^m \ ,$$

which leads to

$$l < \frac{1-\delta r}{r-1}k \ .$$

Thus we have

$$w' = \sum_{k=0}^{m} \left\lfloor \frac{1-\delta r}{r-1}k \right\rfloor \geq \sum_{k=0}^{m} \left(\frac{1-\delta r}{r-1}k + 1 \right) = \frac{1-\delta r}{2(r-1)}m^2 + o(m^2) \ .$$

Finally, combining with Theorem 7 we obtain:

$$\det(L'_y) \leq c \prod_{k=0}^{m} \prod_{l=0}^{\lfloor \frac{1-\delta r}{r-1}k \rfloor} e^{m+(\delta-\frac{1}{r})k+(1-\frac{1}{r})l}$$

$$\det(L'_y) \leq c e^{\frac{(\delta r+3r-1)(1-\delta r)}{6(r-1)r}m^3+o(m^3)} \ ,$$

where c is only a function of δ which can be neglected.

We can now bound

$$\det(L_1) = \det(\Delta)\det(L'_y) < e^{-\frac{(\delta r+r\delta^2-3r+1)}{6(r-1)}m^3+o(m^3)},$$

and we need this term to be lower than $e^{m(w''-1)}$, where $w'' = (1/2)(m+1)(m+2) + w'$, which leads to

$$(-r\delta^2 + 2\delta r - 1)m^3 < 0,$$

which gives

$$\delta < 1 - \frac{\sqrt{r^2 - r}}{r}.$$

For a 3-prime modulus, the obtained bound is around 0.184, which improves the previous one.

5 Concluding Remarks

This report extends Wiener's and Boneh-Durfee's results to the case of multiprime RSA, and shows that these attacks are probably not applicable to Takagi schemes.

As far as multiprime RSA is concerned, results show that, although that attack is still applicable, its efficiency quickly decreases as the number of factors increases. However one must keep in mind that the modulus size has to increase with the number of factors in order to keep the same security level as for classical RSA. In fact the results tend to suggest that there is a fixed range for which the attack is applicable. For example, for a security level comparable to 1024-bit classical RSA the secret exponent must be *theoretically* greater than 250 or 300 bits.

However, we would like to insist on the heuristic nature of LLL and so of the attack, that sometimes turns out to be much more efficient than expected. An extra security margin would therefore be desirable. As Boneh and Durfee pointed out, we cannot give our results as theorems, because nothing ensures that LLL outputs two algebraically independent vectors.

We show in this report that it is possible to use quite a short secret exponent with the RSA MultiPrime. This improves signature generation in comparison with the use of classical RSA and CRT. Nevertheless, as Durfee and Nguyen explain in [7], one should be very cautious when using a short secret exponent with RSA. The bound $N^{1/6}$ is improved by the lattice tools, and it might be possible that it could grow a little if we examine the resolution of modular polynomial equations with low solutions in more details. A way to defeat this attack is to increase the size of e by adding a multiple of $\mathrm{lcm}(p-1,\, q-1,\, r-1)$. Moreover, adding primes in the modulus could make the Boneh-Durfee approach less effective because the number of variables of the polynomials involved in their process could produce too large lattices.

Wiener proposes, as a countermeasure which quickly computes the secret exponentiation, the possibility to find a large secret exponent d such that $d_{p_i} = d \pmod{p_i - 1}$ is small for each prime factor p_i of the modulus. Whether there exists an efficient attack on such secret exponent is an open problem. The best attack known runs in time $\min(\sqrt{d_p},\, \sqrt{d_q})$ for a 2-prime modulus.

6 Acknowledgements

The authors would like to thank Ali Akhavi, Jean-François Misarsky and Francesco Sica for fruitful discussions and useful comments on preliminary versions of this report.

References

[1] Blömer, J., May, A.: Low Secret Exponent RSA Revisited. Cryptography and Lattice Conference (CaLC 2001), Lecture Notes in Computer Science (2001)

[2] Boneh, D., Durfee, G.: Cryptanalysis of RSA with Private Key d Less Than $N^{0.292}$. Advances in Cryptology - Proceedings of Eurocrypt '99, Lecture Notes in Computer Science **1952** (1999) 1–11

[3] Boneh, D., Durfee, G.: Cryptanalysis of RSA with Private Key d Less Than $N^{0.292}$. IEEE Transactions on Information Theory **46:4** (2000) 1339–1349

[4] Boneh, D.: Twenty Years of Attacks on the RSA Cryptosystem. Notices of the American Mathematical Society **46:2** (1999) 203–213

[5] Boneh, D. and Durfee, G. and Howgrave-Graham, N.: Factoring $N = p^r q$ for Large r. Advances in Cryptology – Proceedings of CRYPTO 1999, Lecture Notes in Computer Science **1666** (1999) 326–337

[6] Coppersmith, D.: Small Solutions to Polynomial Equations, and Low Exponent RSA Vulnerabilities. Journal of Cryptology **10:4** (1997) 233–260

[7] Durfee, G., Nguyen, P. Q.: Cryptanalysis of RSA Schemes with Short Secret Exponent from Asiacrypt '99. Advances in Cryptology - Proceedings of Asiacrypt '00, Lecture Notes in Computer Science **1976** (2000) 14–29

[8] Ebinger, P. and Teske, E.: Factoring $N = pq^2$ with the Elliptic Curve Method. Advances in Cryptology – Proceedings of ANTS-V, Lecture Notes in Computer Science **2369** (2002) 475–490

[9] Hardy, G. H., Wright E. M.: An Introduction to the Theory of Numbers – Fifth Edition. Oxford Science Publications (1979)

[10] MJ. Hinek, MK. Low, E. Teske: On Some Attacks on Multi-prime RSA. Advances in Cryptology - Proceedings of SAC 2002, Lecture Notes in Computer Science , To Appear

[11] Howgrave-Graham, N.: Finding Small Roots of Univariate Modular Equations Revisited. Cryptography and coding, Lecture Notes in Computer Science **1355** (1997) 131–142

[12] Lenstra, A.K., Lenstra Jr., H.W., Lovász, L.: Factoring Polynomials with Rational Coefficients. Mathematische Annalen **261** (1982) 515–534

[13] May, A.: Cryptanalysis of Unbalanced RSA with Small CRT-Exponent. Advances in Cryptology – Proceedings of Crypto 2002, Lecture Notes in Computer Sciences **2442** (2002) 242–256

[14] Qiao, G., Lam K.-Y.: RSA Signature Algorithm for Microcontroller Implementation. Advances in Cryptology - Proceedings of CARDIS 1998, Lecture Notes in Computer Science **1820** (2000) 353–356

[15] Quisquater, J.-J., Couvreur C.: Fast Decipherment Algorithm for RSA Public-Key Cryptosystem. Electronics Letters **18:21** (1982) 905–907

[16] Rivest, R.L., Shamir, A., Adleman, L.: A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. Technical Report LCS!TM82, MIT Laboratory for Computer Science, Cambridge, Massachusetts (1977)

[17] Shamir, A.: RSA for Paranoids. RSA Laboratories' CryptoBytes **1:3** (1995) 1–4

[18] Strassen, V.: Einige Resultate über Berechnungskomplexität. Jahresberichte Deutscher Math. Vereinigung, (1976/77) 1–8

[19] Sun, H.-M., Yang, W.-C., Laih., C.-S.: On the Design of RSA with Short Secret Exponent. Advances in Cryptology - Proceedings of Asiacrypt '99, Lecture Notes in Computer Science **1716** (1999) 150–164

[20] Takagi, T.: Fast RSA-Type Cryptosystems Using N-Adic Expansion in Advances in Cryptography - Proceedings of CRYPTO 1997, pp. 372-384, volume 1294, Lecture Notes in Computer Science series, (1997).

[21] Takagi, T.: Fast RSA-Type Cryptosystem Modulo $p^k q$ in Advances in Cryptography - Proceedings of CRYPTO 1998, pp. 318-326, volume 1462, Lecture Notes in Computer Science series, (1998).

[22] Takagi, T.: New public-key cryptosystems with fast decryption. Dissertation, PhD thesis (2001)

[23] Lim, S., Kim, S., Yie, I., Lee, H.: A Generalized Takagi-Cryptosystem with a modulus of the form $p^r q^s$ in Advances in Cryptography - Proceedings of Indocrypt 1998, pp. 283-294, volume 1977, Lecture Notes in Computer Science series, (2000).

[24] Verheul, E. R., van Tilborg, H. C. A.: Cryptanalysis of 'Less Short' RSA Secret Exponents. Applicable Algebra in Engineering, Communication and Computing **8** (1997) 425–435

[25] Wiener, M. J.: Cryptanalysis of Short RSA Secret Exponents. IEEE Transaction on Information Theory **36:3** (1990) 553–558

[26] Compaq Computer Corporation: Cryptography Using Compaq Multi-Prime Technology in a Parallel Processing Environment. (2000) Available at: http://nonstop.compaq.com/view.asp?IOID=4523

A Lattice Attacks

A.1 Adaptation of lattice attack

In [23], it was suggested to use a modulus such that $r = s + 1$, in this case the explicit bound is given by: for $N = p^{s+1}q^s$, if there exists d' as before, such that

$$d' < N^{(\frac{8}{3}s+1-\frac{2}{3}\sqrt{16s^2+6s})/(2s+1)}$$

then its exact value can be recovered in polynomial time.

Figure 3 illustrates numerical results for upper bound.

N	p^2q	p^3q^2	p^4q^3	p^5q^4	p^6q^5	p^7q^6	p^8q^7
Upper bound	0.1799	0.1043	0.0735	0.0568	0.0463	0.0390	0.0338

Figure 3: Upper bound on δ for Boneh-Durfee's attack, when constants are neglected

In the following, we apply Hinek and al.'s correction to Boneh-Durfee's attack.

Considering that e has the same size as N and that $d = N^\delta$, we easily get

$$|k| = |\tfrac{1-ed}{\varphi(N)}| < 2e^\delta \tag{32}$$

$$|u| < |2^s p^{r+s-1}| < |2^s N^{\frac{r+s-1}{r+s}}| \tag{33}$$

The exact expression of the upper bound on δ is too complex to fit in these pages[2]. They also present a tabular. Figure 4 presents numerical results for some typical values of r, s and N.

N	1024	2048	3072	8192
p^2q	0.163	0.169	0.170	n/a
p^3q^2	0.090	0.095	0.097	n/a
p^4q^3	0.058	0.065	0.067	n/a
p^8q^7				0.030

Figure 4: Upper bound on δ for various modulus forms and sizes

A.2 Geometrically progressive matrices

Once again, it is interesting to see how this applies to the suggestion of [23] to use a modulus such that $r = s+1$. In this case the explicit bound is given by: for $N = p^{s+1}q^s$, if

$$d' < N^{1-\sqrt{\frac{2s}{2s+1}}}$$

then its exact value can be recovered in polynomial time.

Figure 5 gives numerical results.

N	p^2q	p^3q^2	p^4q^3	p^5q^4	p^6q^5	p^7q^6	p^8q^7
Boneh-Durfee	0.1835	0.1056	0.0742	0.0572	0.0465	0.0392	0.0339

Figure 5: Upper bound on δ for Boneh-Durfee's geometrical progressive matrices attack

B Summary

Figure 6 gives a table to summarize the bound obtained in this report.

N	p^2q	p^3q^2	p^4q^3	p^5q^4	p^6q^5	p^7q^6	p^8q^7
Wiener	0.1667	0.1000	0.0714	0.0556	0.0455	0.0385	0.0333
Boneh-Durfee	0.1799	0.1043	0.0735	0.0568	0.0463	0.0390	0.0338
B-D's geo. proj.	0.1835	0.1056	0.0742	0.0572	0.0465	0.0392	0.0339

Figure 6: Upper bound on δ for Wiener's attack, and Boneh-Durfee's attack and Boneh-Durfee's geometrical progressive matrices attack

[2]Note that Hinek et al. experience the same problem: although Boneh-Durfee's initial (incorrect) bound is elegant ($\delta = 0.292$), the corrected bound is more complex and depending on N.